

Nathan Wemmer

Code ▼

This is an R Markdown (<http://rmarkdown.rstudio.com>) Notebook. When you execute code within the notebook, the results appear beneath the code.

Try executing this chunk by clicking the *Run* button within the chunk or by placing your cursor inside it and pressing *Ctrl+Shift+Enter*.

Add a new chunk by clicking the *Insert Chunk* button on the toolbar or by pressing *Ctrl+Alt+I*.

When you save the notebook, an HTML file containing the code and output will be saved alongside it (click the *Preview* button or press *Ctrl+Shift+K* to preview the HTML file).

The preview shows you a rendered HTML copy of the contents of the editor. Consequently, unlike *Knit*, *Preview* does not run any R code chunks. Instead, the output of the chunk when it was last run in the editor is displayed. # Chapter 6

Section 6.1

You told us to skip this one, since the data file was not included. # Section 6.2

Hide

```
data<-read.table("Daphnia.txt",header=T)
```

```
attach(data)
names(data)
```

```
[1] "Growth.rate" "Water"      "Detergent"  "Daphnia"
```

Hide

```
tapply(Growth.rate,Detergent,mean)
```

```
BrandA BrandB BrandC BrandD
3.884832 4.010044 3.954512 3.558231
```

Hide

```
tapply(Growth.rate,Water,mean)
```

```
Tyne Wear
3.685862 4.017948
```

Hide

```
tapply(Growth.rate,Daphnia,mean)
```

Clone1	Clone2	Clone3
2.839875	4.577121	4.138719

Hide

```
tapply(Growth.rate,list(Daphnia,Detergent),mean)
```

	BrandA	BrandB	BrandC	BrandD
Clone1	2.732227	2.929140	3.071335	2.626797
Clone2	3.919002	4.402931	4.772805	5.213745
Clone3	5.003268	4.698062	4.019397	2.834151

Hide

```
tapply(Growth.rate,list(Daphnia,Detergent),median)
```

	BrandA	BrandB	BrandC	BrandD
Clone1	2.705995	3.012495	3.073964	2.503468
Clone2	3.924411	4.282181	4.612801	5.416785
Clone3	5.057594	4.627812	4.040108	2.573003

Hide

```
tapply(Growth.rate,list(Daphnia,Detergent), function(x) sqrt(var(x)/length(x)))
```

	BrandA	BrandB	BrandC	BrandD
Clone1	0.2163448	0.2319320	0.3055929	0.1905771
Clone2	0.4702855	0.3639819	0.5773096	0.5520220
Clone3	0.2688604	0.2683660	0.5395750	0.4260212

Hide

```
tapply(Growth.rate,list(Daphnia,Detergent,Water),mean)
```

```
, , Tyne
```

	BrandA	BrandB	BrandC	BrandD
Clone1	2.811265	2.775903	3.287529	2.597192
Clone2	3.307634	4.191188	3.620532	4.105651
Clone3	4.866524	4.766258	4.534902	3.365766

```
, , Wear
```

	BrandA	BrandB	BrandC	BrandD
Clone1	2.653189	3.082377	2.855142	2.656403
Clone2	4.530371	4.614673	5.925078	6.321838
Clone3	5.140011	4.629867	3.503892	2.302537

Hide

```
ftable(tapply(Growth.rate,list(Daphnia,Detergent,Water),mean))
```

		Tyne	Wear
Clone1	BrandA	2.811265	2.653189
	BrandB	2.775903	3.082377
	BrandC	3.287529	2.855142
	BrandD	2.597192	2.656403
Clone2	BrandA	3.307634	4.530371
	BrandB	4.191188	4.614673
	BrandC	3.620532	5.925078
	BrandD	4.105651	6.321838
Clone3	BrandA	4.866524	5.140011
	BrandB	4.766258	4.629867
	BrandC	4.534902	3.503892
	BrandD	3.365766	2.302537

[Hide](#)

```
water<-factor(Water,levels=c("Wear","Tyne"))
ftable(tapply(Growth.rate,list(Daphnia,Detergent,water),mean))
```

		Wear	Tyne
Clone1	BrandA	2.653189	2.811265
	BrandB	3.082377	2.775903
	BrandC	2.855142	3.287529
	BrandD	2.656403	2.597192
Clone2	BrandA	4.530371	3.307634
	BrandB	4.614673	4.191188
	BrandC	5.925078	3.620532
	BrandD	6.321838	4.105651
Clone3	BrandA	5.140011	4.866524
	BrandB	4.629867	4.766258
	BrandC	3.503892	4.534902
	BrandD	2.302537	3.365766

[Hide](#)

```
tapply(Growth.rate,Detergent,mean,trim=0.1)
```

BrandA	BrandB	BrandC	BrandD
3.874869	4.019206	3.890448	3.482322

[Hide](#)

```
tapply(Growth.rate,Detergent,mean,na.rm=T)
```

```
BrandA BrandB BrandC BrandD
3.884832 4.010044 3.954512 3.558231
```

Hide

```
dets <- as.vector(tapply(as.numeric(Detergent),list(Detergent,Daphnia),mean))
levels(Detergent)[dets]
```

```
[1] "BrandA" "BrandB" "BrandC" "BrandD" "BrandA" "BrandB"
[7] "BrandC" "BrandD" "BrandA" "BrandB" "BrandC" "BrandD"
```

Hide

```
clones<-as.vector(tapply(as.numeric(Daphnia),list(Detergent,Daphnia),mean))
levels(Daphnia)[clones]
```

```
[1] "Clone1" "Clone1" "Clone1" "Clone1" "Clone2" "Clone2"
[7] "Clone2" "Clone2" "Clone3" "Clone3" "Clone3" "Clone3"
```

Hide

```
tapply(Growth.rate,list(Detergent,Daphnia),mean)
```

```
      Clone1 Clone2 Clone3
BrandA 2.732227 3.919002 5.003268
BrandB 2.929140 4.402931 4.698062
BrandC 3.071335 4.772805 4.019397
BrandD 2.626797 5.213745 2.834151
```

Hide

```
means <- as.vector(tapply(Growth.rate,list(Detergent,Daphnia),mean))
detergent <- levels(Detergent)[dets]
daphnia <- levels(Daphnia)[clones]
data.frame(means,detergent,daphnia)
```

	means <dbl>	detergent <fctr>	daphnia <fctr>
	2.732227	BrandA	Clone1
	2.929140	BrandB	Clone1
	3.071335	BrandC	Clone1
	2.626797	BrandD	Clone1
	3.919002	BrandA	Clone2
	4.402931	BrandB	Clone2

means <dbl>	detergent <fctr>	daphnia <fctr>
4.772805	BrandC	Clone2
5.213745	BrandD	Clone2
5.003268	BrandA	Clone3
4.698062	BrandB	Clone3

1-10 of 12 rows

Previous **1** 2 Next

Hide

```
as.data.frame(table(tapply(Growth.rate,list(Detergent,Daphnia),mean)))
```

Var1 <fctr>	Var2 <fctr>	Freq <dbl>
BrandA	Clone1	2.732227
BrandB	Clone1	2.929140
BrandC	Clone1	3.071335
BrandD	Clone1	2.626797
BrandA	Clone2	3.919002
BrandB	Clone2	4.402931
BrandC	Clone2	4.772805
BrandD	Clone2	5.213745
BrandA	Clone3	5.003268
BrandB	Clone3	4.698062

1-10 of 12 rows

Previous **1** 2 Next

Hide

```
new<-as.data.frame(table(tapply(Growth.rate,list(Detergent,Daphnia),mean)))
names(new)<-c("detergents","daphina","means")
head(new)
```

	detergents <fctr>	daphina <fctr>	means <dbl>
1	BrandA	Clone1	2.732227
2	BrandB	Clone1	2.929140

	detergents <fctr>	daphina <fctr>	means <dbl>
3	BrandC	Clone1	3.071335
4	BrandD	Clone1	2.626797
5	BrandA	Clone2	3.919002
6	BrandB	Clone2	4.402931
6 rows			

Hide

NA
NA
NA
NA

Section 6.3

Hide

```
count.table<-read.table("tabledata.txt",header=T)
attach(count.table)
head(count.table)
```

	count <int>	sex <fctr>	age <fctr>	condition <fctr>
1	12	male	young	healthy
2	7	male	old	healthy
3	9	female	young	healthy
4	8	female	old	healthy
5	6	male	young	parasitized
6	7	male	old	parasitized
6 rows				

Hide

```
lapply(count.table,function(x)rep(x, count.table$count))
```

\$count

```
[1] 12 12 12 12 12 12 12 12 12 12 12 12 12 7 7 7 7 7 7 7
[20] 9 9 9 9 9 9 9 9 9 9 8 8 8 8 8 8 8 8 8 6 6
[39] 6 6 6 6 7 7 7 7 7 7 8 8 8 8 8 8 8 8 8 8 8
[58] 5 5 5 5 5
```

\$sex

```
[1] male male male male male male male male
[9] male male male male male male male male
[17] male male male female female female female female
[25] female female female female female female female female
[33] female female female female male male male male
[41] male male male male male male male male
[49] male female female female female female female female
[57] female female female female female female
Levels: female male
```

\$age

```
[1] young young young young young young young young young
[10] young young young old old old old old old
[19] old young young young young young young young young
[28] young old old old old old old old old
[37] young young young young young young old old old
[46] old old old old young young young young young
[55] young young young old old old old old
Levels: old young
```

\$condition

```
[1] healthy healthy healthy healthy
[5] healthy healthy healthy healthy
[9] healthy healthy healthy healthy
[13] healthy healthy healthy healthy
[17] healthy healthy healthy healthy
[21] healthy healthy healthy healthy
[25] healthy healthy healthy healthy
[29] healthy healthy healthy healthy
[33] healthy healthy healthy healthy
[37] parasitized parasitized parasitized parasitized
[41] parasitized parasitized parasitized parasitized
[45] parasitized parasitized parasitized parasitized
[49] parasitized parasitized parasitized parasitized
[53] parasitized parasitized parasitized parasitized
[57] parasitized parasitized parasitized parasitized
[61] parasitized parasitized
Levels: healthy parasitized
```

Hide

```
dbtable<-as.data.frame(lapply(count.table,function(x)rep(x, count.table$count)))
head(dbtable)
```

	count <int>	sex <fctr>	age <fctr>	condition <fctr>
1	12	male	young	healthy
2	12	male	young	healthy
3	12	male	young	healthy
4	12	male	young	healthy
5	12	male	young	healthy
6	12	male	young	healthy

6 rows

Hide

```
dbtable<-dbtable[, -1]  
head(dbtable)
```

	sex <fctr>	age <fctr>	condition <fctr>
1	male	young	healthy
2	male	young	healthy
3	male	young	healthy
4	male	young	healthy
5	male	young	healthy
6	male	young	healthy

6 rows

Hide

```
tail(dbtable)
```

	sex <fctr>	age <fctr>	condition <fctr>
57	female	young	parasitized
58	female	old	parasitized
59	female	old	parasitized
60	female	old	parasitized
61	female	old	parasitized

	sex <fctr>	age <fctr>	condition <fctr>
62	female	old	parasitized
6 rows			

Hide

NA

Section 6.4

Hide

```
table(dbtable)
```

```
, , condition = healthy
```

```
      age
sex    old young
female  8     9
male    7    12
```

```
, , condition = parasitized
```

```
      age
sex    old young
female  5     8
male    7     6
```

Hide

```
as.data.frame(table(dbtable))
```

sex <fctr>	age <fctr>	condition <fctr>	Freq <int>
female	old	healthy	8
male	old	healthy	7
female	young	healthy	9
male	young	healthy	12
female	old	parasitized	5
male	old	parasitized	7
female	young	parasitized	8
male	young	parasitized	6

8 rows

Hide

```
frame<-as.data.frame(table(dtable))
names(frame)[4]<-"count"
frame
```

sex <fctr>	age <fctr>	condition <fctr>	count <int>
female	old	healthy	8
male	old	healthy	7
female	young	healthy	9
male	young	healthy	12
female	old	parasitized	5
male	old	parasitized	7
female	young	parasitized	8
male	young	parasitized	6

8 rows

Hide

NA

Section 6.5

Hide

```
counts<-matrix(c(2,2,4,3,1,4,2,0,1,5,3,3),nrow=4)
counts
```

```
      [,1] [,2] [,3]
[1,]    2    1    1
[2,]    2    4    5
[3,]    4    2    3
[4,]    3    0    3
```

Hide

```
prop.table(counts,1)
```

```

      [,1]      [,2]      [,3]
[1,] 0.5000000 0.2500000 0.2500000
[2,] 0.1818182 0.3636364 0.4545455
[3,] 0.4444444 0.2222222 0.3333333
[4,] 0.5000000 0.0000000 0.5000000

```

Hide

```
prop.table(counts,2)
```

```

      [,1]      [,2]      [,3]
[1,] 0.1818182 0.1428571 0.08333333
[2,] 0.1818182 0.5714286 0.41666667
[3,] 0.3636364 0.2857143 0.25000000
[4,] 0.2727273 0.0000000 0.25000000

```

Hide

```
colSums(prop.table(counts,2))
```

```
[1] 1 1 1
```

Hide

```
prop.table(counts)
```

```

      [,1]      [,2]      [,3]
[1,] 0.06666667 0.03333333 0.03333333
[2,] 0.06666667 0.13333333 0.16666667
[3,] 0.13333333 0.06666667 0.10000000
[4,] 0.10000000 0.00000000 0.10000000

```

Hide

```
sum(prop.table(counts))
```

```
[1] 1
```

Section 6.6

Hide

```
scale(counts)
```

```
      [,1]      [,2]      [,3]
[1,] -0.7833495 -0.439155 -1.224745
[2,] -0.7833495  1.317465  1.224745
[3,]  1.3055824  0.146385  0.000000
[4,]  0.2611165 -1.024695  0.000000
attr(,"scaled:center")
[1] 2.75 1.75 3.00
attr(,"scaled:scale")
[1] 0.9574271 1.7078251 1.6329932
```

Hide

```
apply(counts,2,sd)
```

```
[1] 0.9574271 1.7078251 1.6329932
```

Section 6.7

Hide

```
expand.grid(height = seq(60, 80, 5), weight = seq(100, 300, 50), sex = c("Male","Female"))
```

height	weight	sex
<dbl>	<dbl>	<fctr>
60	100	Male
65	100	Male
70	100	Male
75	100	Male
80	100	Male
60	150	Male
65	150	Male
70	150	Male
75	150	Male
80	150	Male

1-10 of 50 rows

Previous12345Next

Hide

```
NA
```

Section 6.8

Hide

```
data<-read.table("parasites.txt")
names(data)<-"parasite"
attach(data)
head(data)
```

parasite
<fctr>

1	vulgaris
2	splendens
3	knowlesii
4	vulgaris
5	knowlesii
6	viridis

6 rows

Hide

```
levels(parasite)
```

```
[1] "knowlesii" "kochii"      "splendens" "viridis"    "vulgaris"
```

Hide

```
vulgaris<-factor(1*(parasite=="vulgaris"))
kochii<-factor(1*(parasite=="kochii"))
table(vulgaris)
```

```
vulgaris
 0  1
99 52
```

Hide

```
table(kochii)
```

```
kochii
 0  1
134 17
```

Hide

```
model.matrix(~parasite-1)
```

	parasiteknowlesii	parasitekochii	parasitesplendens	parasiteviridis	parasitevulgaris
1	0	0	0	0	1
2	0	0	1	0	0
3	1	0	0	0	0
4	0	0	0	0	1
5	1	0	0	0	0
6	0	0	0	1	0
7	0	0	1	0	0
8	0	0	1	0	0
9	0	0	0	1	0
10	0	0	0	0	1
11	0	0	1	0	0
12	0	0	0	1	0
13	0	0	0	1	0
14	0	1	0	0	0
15	0	1	0	0	0
16	0	0	0	0	1
17	1	0	0	0	0
18	0	0	1	0	0
19	0	0	0	0	1
20	0	0	1	0	0
21	0	0	0	0	1
22	1	0	0	0	0
23	0	0	1	0	0
24	0	0	0	0	1
25	0	0	0	0	1
26	0	0	0	1	0
27	0	0	1	0	0
28	0	0	1	0	0
29	0	0	0	0	1
30	0	0	0	0	1
31	1	0	0	0	0
32	1	0	0	0	0
33	0	0	1	0	0
34	0	0	1	0	0
35	0	0	0	0	1
36	0	0	0	1	0
37	0	0	0	0	1
38	0	1	0	0	0
39	0	0	0	1	0
40	0	0	0	1	0
41	0	0	1	0	0
42	0	1	0	0	0
43	0	0	0	0	1
44	1	0	0	0	0
45	0	0	0	0	1
46	0	0	0	0	1
47	0	1	0	0	0
48	0	0	0	0	1
49	1	0	0	0	0
50	0	0	0	0	1
51	0	1	0	0	0
52	0	0	1	0	0

53	0	0	1	0	0
54	0	0	0	0	1
55	0	0	1	0	0
56	0	0	0	1	0
57	0	0	0	0	1
58	0	0	1	0	0
59	0	0	0	1	0
60	0	0	0	1	0
61	0	0	1	0	0
62	0	0	0	1	0
63	0	0	0	0	1
64	0	0	0	0	1
65	0	0	0	0	1
66	0	0	0	1	0
67	0	0	0	1	0
68	0	0	0	0	1
69	1	0	0	0	0
70	0	0	0	1	0
71	0	0	0	0	1
72	0	0	1	0	0
73	0	0	1	0	0
74	0	0	0	0	1
75	0	0	0	0	1
76	0	0	0	0	1
77	0	0	0	1	0
78	0	0	0	0	1
79	0	0	0	0	1
80	0	0	0	0	1
81	0	1	0	0	0
82	0	0	0	0	1
83	0	1	0	0	0
84	0	0	1	0	0
85	1	0	0	0	0
86	0	0	1	0	0
87	0	0	0	0	1
88	0	0	0	0	1
89	0	0	1	0	0
90	0	0	0	0	1
91	0	0	0	0	1
92	0	0	0	0	1
93	0	0	1	0	0
94	0	0	1	0	0
95	0	0	0	1	0
96	0	1	0	0	0
97	0	0	0	1	0
98	0	0	1	0	0
99	0	0	0	1	0
100	0	0	1	0	0
101	0	0	0	0	1
102	0	1	0	0	0
103	0	0	0	0	1
104	0	0	0	0	1
105	0	0	0	0	1
106	0	1	0	0	0

107	0	1	0	0	0
108	0	0	1	0	0
109	0	0	0	0	1
110	1	0	0	0	0
111	1	0	0	0	0
112	0	0	0	1	0
113	1	0	0	0	0
114	0	0	1	0	0
115	0	0	0	0	1
116	0	0	0	1	0
117	0	0	1	0	0
118	0	0	0	0	1
119	1	0	0	0	0
120	1	0	0	0	0
121	1	0	0	0	0
122	0	0	0	0	1
123	1	0	0	0	0
124	0	0	0	1	0
125	1	0	0	0	0
126	0	0	0	0	1
127	0	1	0	0	0
128	0	0	0	1	0
129	0	0	0	1	0
130	1	0	0	0	0
131	0	0	0	1	0
132	1	0	0	0	0
133	0	0	0	0	1
134	0	0	0	0	1
135	0	1	0	0	0
136	0	0	0	0	1
137	0	0	0	1	0
138	0	0	0	0	1
139	1	0	0	0	0
140	0	0	0	1	0
141	0	0	0	0	1
142	0	1	0	0	0
143	0	0	0	0	1
144	0	0	1	0	0
145	0	0	0	1	0
146	0	1	0	0	0
147	1	0	0	0	0
148	0	0	0	1	0
149	0	0	0	0	1
150	0	1	0	0	0
151	0	0	1	0	0

```
attr("assign")
```

```
[1] 1 1 1 1 1
```

```
attr("contrasts")
```

```
attr("contrasts")$parasite
```

```
[1] "contr.treatment"
```

[Hide](#)


```
# not work! where is original.frame?
new.frame<-data.frame(original.frame, model.matrix(~parasite-1))
```

```
Error in data.frame(original.frame, model.matrix(~parasite - 1)) :
  object 'original.frame' not found
```

Section 6.9

Hide

```
table(c(2,2,2,7,7,11))
```

```
2  7 11
3  2  1
```

Hide

```
tabulate(c(2,2,2,7,7,11))
```

```
[1] 0 3 0 0 0 0 2 0 0 0 1
```

Hide

```
tabulate(c(2,0,-3,2,2,7,-1, 0,0,7,11))
```

```
[1] 0 3 0 0 0 0 2 0 0 0 1
```

Hide

```
table(rnbinom(100,1,0.2))
```

```
0  1  2  3  4  5  6  7  8  9 10 11 13 14 16 17
16 20 14 10  8  7  6  5  4  3  1  1  1  1  1  2
```

Hide

```
table(rnbinom(100,1,0.2))
```

```
0  1  2  3  4  5  6  7  8  9 11 12 13 14 16 20 21 22
14 18  6  8  8 12  9  4  5  3  5  1  1  2  1  1  1  1
```

Hide

```
table(rnbinom(100,1,0.2))
```

```
 0  1  2  3  4  5  6  7  8 10 11 14 18 19 20 23
16 14 15 10 15  7  6  2  3  1  1  5  1  1  1  2
```

Hide

```
tabulate(rnbinom(100,1,0.2)+1,30)
```

```
[1] 28 16  9  9 10  3  7  2  2  2  2  2  0  4  1  1  0  1  0  1  0  0  0  0  0  0  0  0  0  0
```

Hide

```
totals<-numeric(1000)
for (i in 1:1000) totals[i] <- sum(tabulate(rnbinom(100,1,0.2)+1,30))
table(totals)
```

```
totals
 98  99 100
 5 114 881
```

Chapter 7

Section 7.1.1

Hide

```
x <- seq(0,10,0.1)

windows(7,4)
par(mfrow=c(1,2))
y <- exp(x)
plot(y~x,type="l",main="Exponential")
y <- log(x)
plot(y~x,type="l",main="Logarithmic")
```

Hide

```
NA
NA
```

Section 7.1.2

Hide

```
windows(7,7)
par(mfrow=c(2,2))
x <- seq(0,2*pi,2*pi/100)
y1 <- cos(x)
y2 <- sin(x)
y3 <- tan(x)
plot(y1~x,type="l",main="cosine")
plot(y2~x,type="l",main="sine")
```

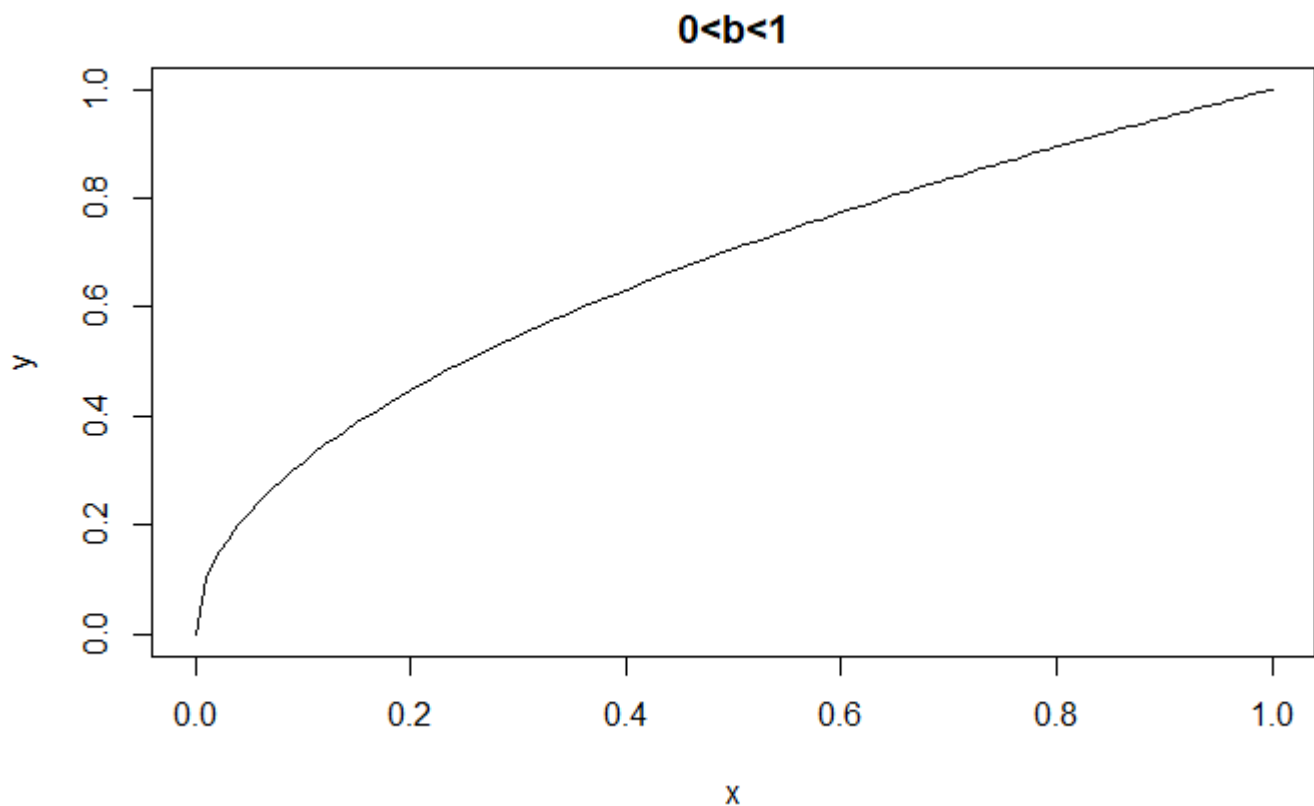
[Hide](#)

```
plot(y3~x,type="l",ylim=c(-3,3),main="tangent")
```

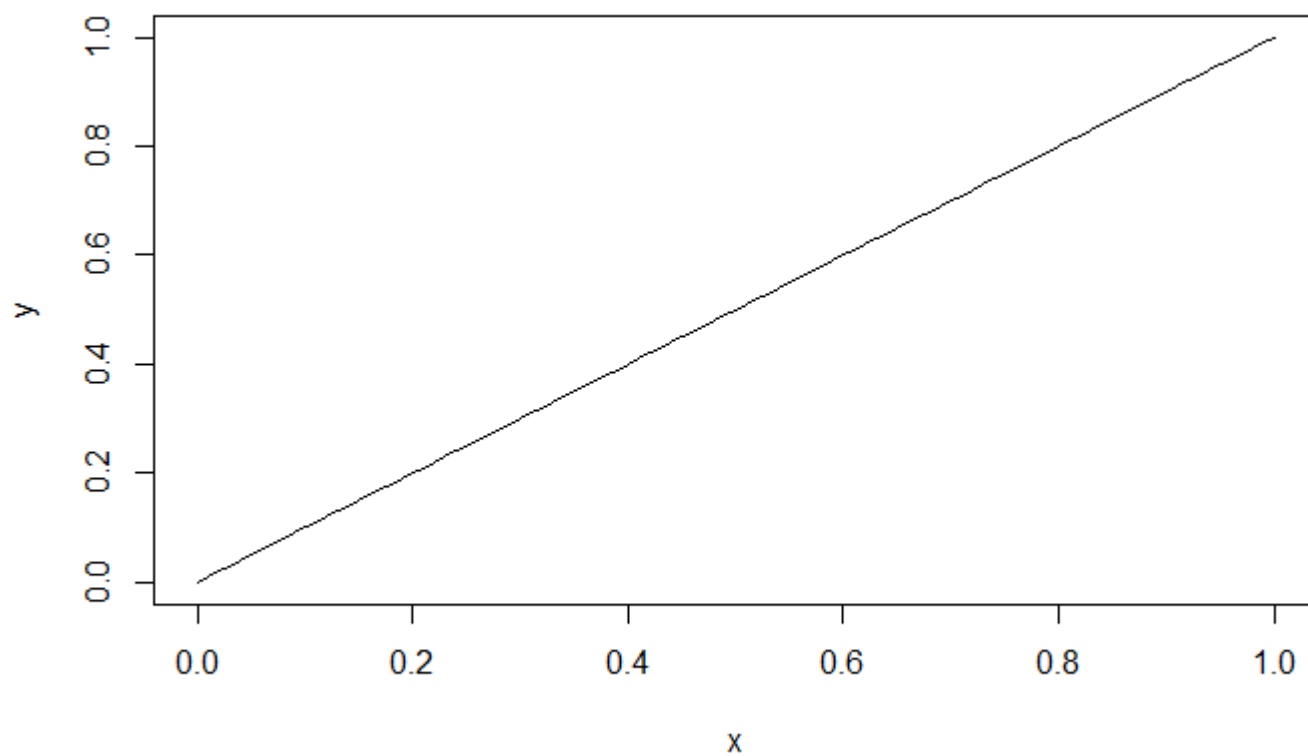
Section 7.1.3

[Hide](#)

```
x <- seq(0,1,0.01)
y <- x^0.5
plot(x,y,type="l",main="0<b<1")
```

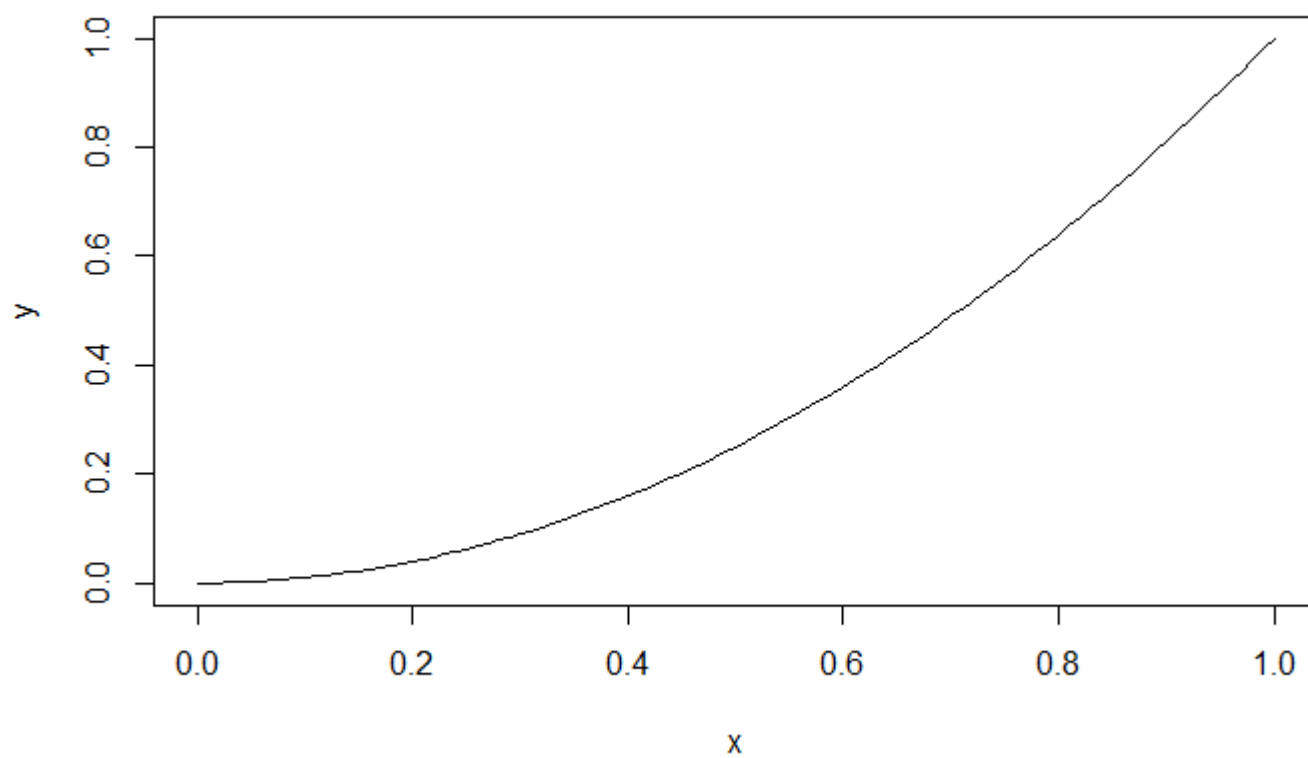
[Hide](#)

```
y <- x
plot(x,y,type="l",main="b=1")
```

b=1

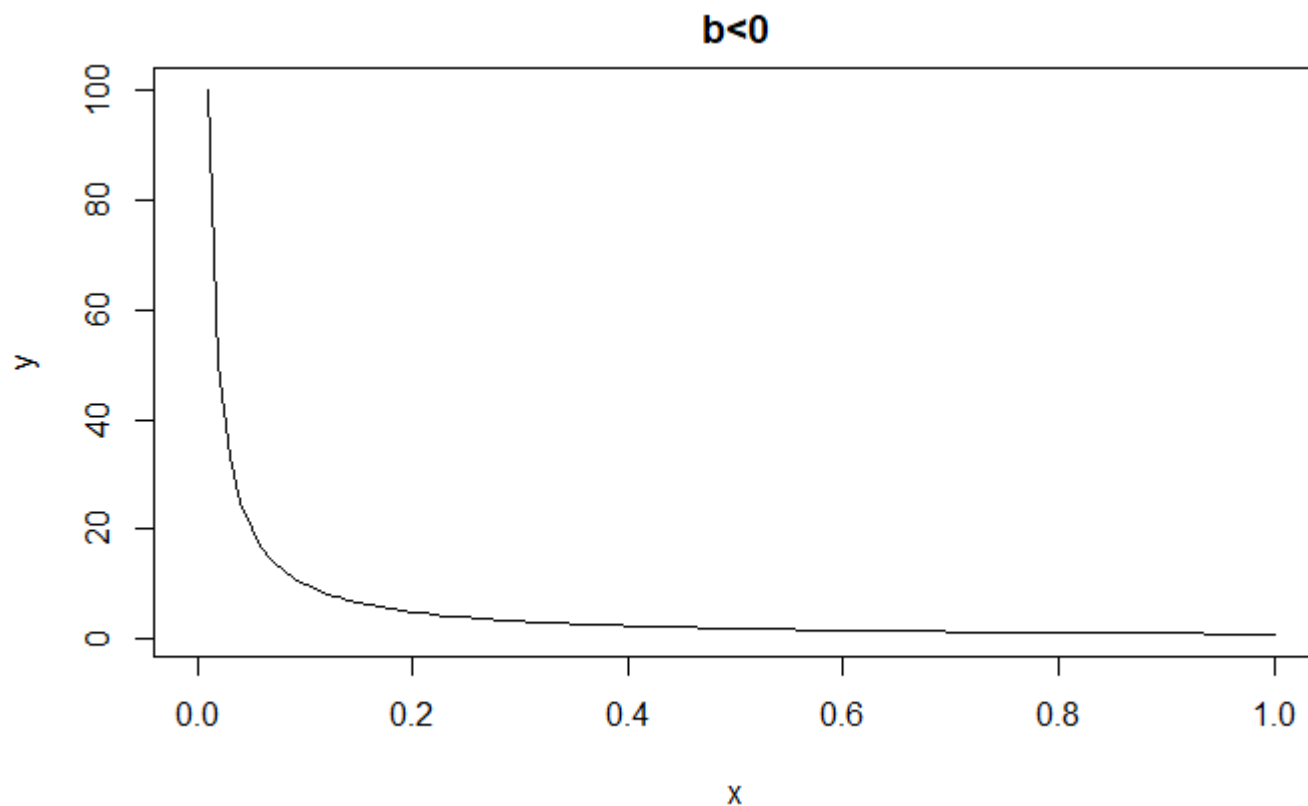
Hide

```
y <- x^2  
plot(x,y,type="l",main="b>1")
```

b>1

Hide

```
y <- 1/x
plot(x,y,type="l",main="b<0")
```



Section 7.1.4

Hide

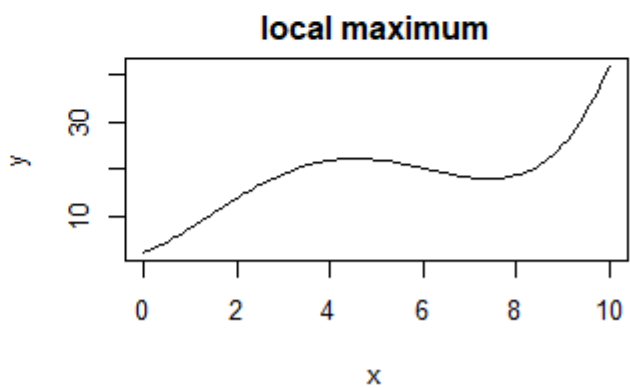
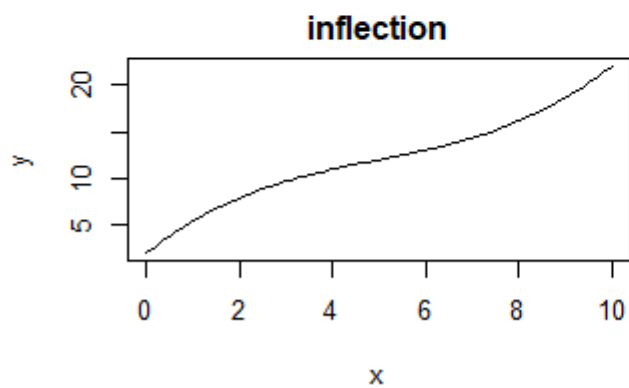
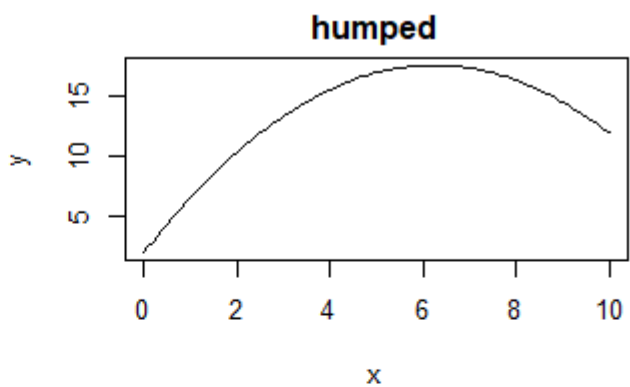
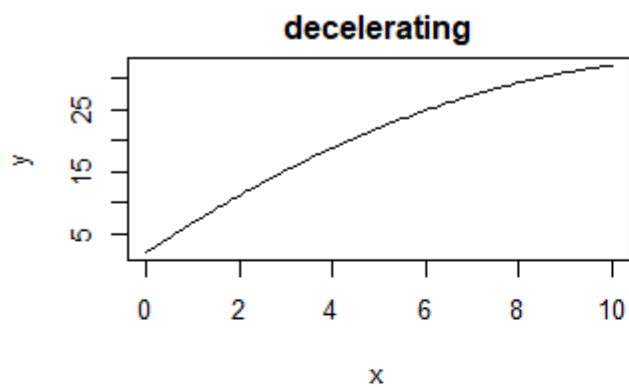
```
x <- seq(0,10,0.1)
y1 <- 2+5*x-0.2*x^2
y2 <- 2+5*x-0.4*x^2
y3 <- 2+4*x-0.6*x^2+0.04*x^3
y4 <- 2+4*x+2*x^2-0.6*x^3+0.04*x^4
par(mfrow=c(2,2))
plot(x,y1,type="l",ylab="y",main="decelerating")
plot(x,y2,type="l",ylab="y",main="humped")
```

Hide

```
plot(x,y3,type="l",ylab="y",main="inflection")
plot(x,y4,type="l",ylab="y",main="local maximum")
```

Hide

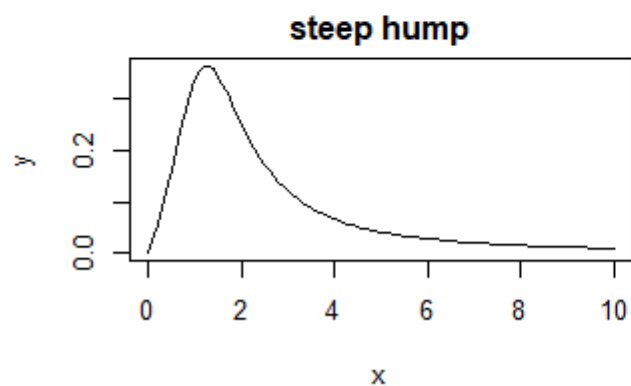
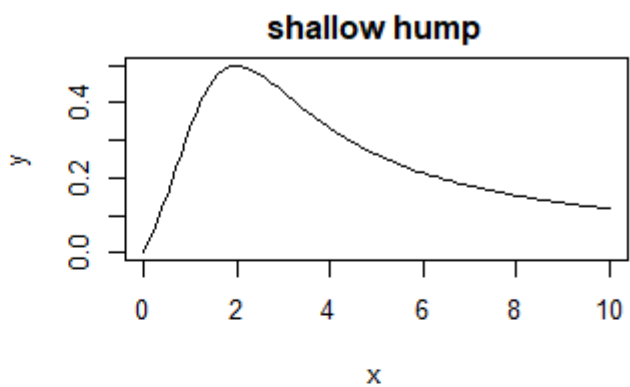
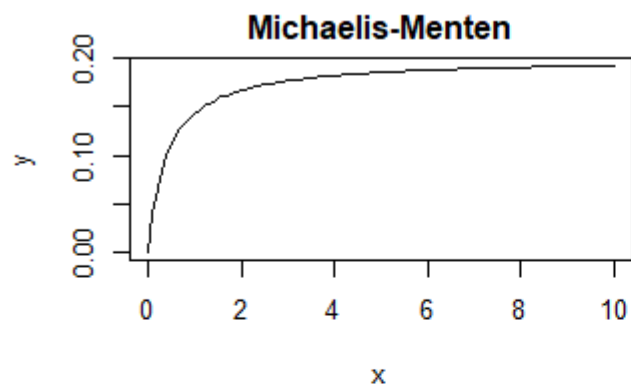
```
par(mfrow=c(2,2))
```


[Hide](#)

```
y1 <- x/(2+5*x)
y2 <- 1/(x-2+4/x)
y3 <- 1/(x^2-2+4/x)
plot(x,y1,type="l",ylab="y",main="Michaelis-Menten")
plot(x,y2,type="l",ylab="y",main="shallow hump")
```

[Hide](#)

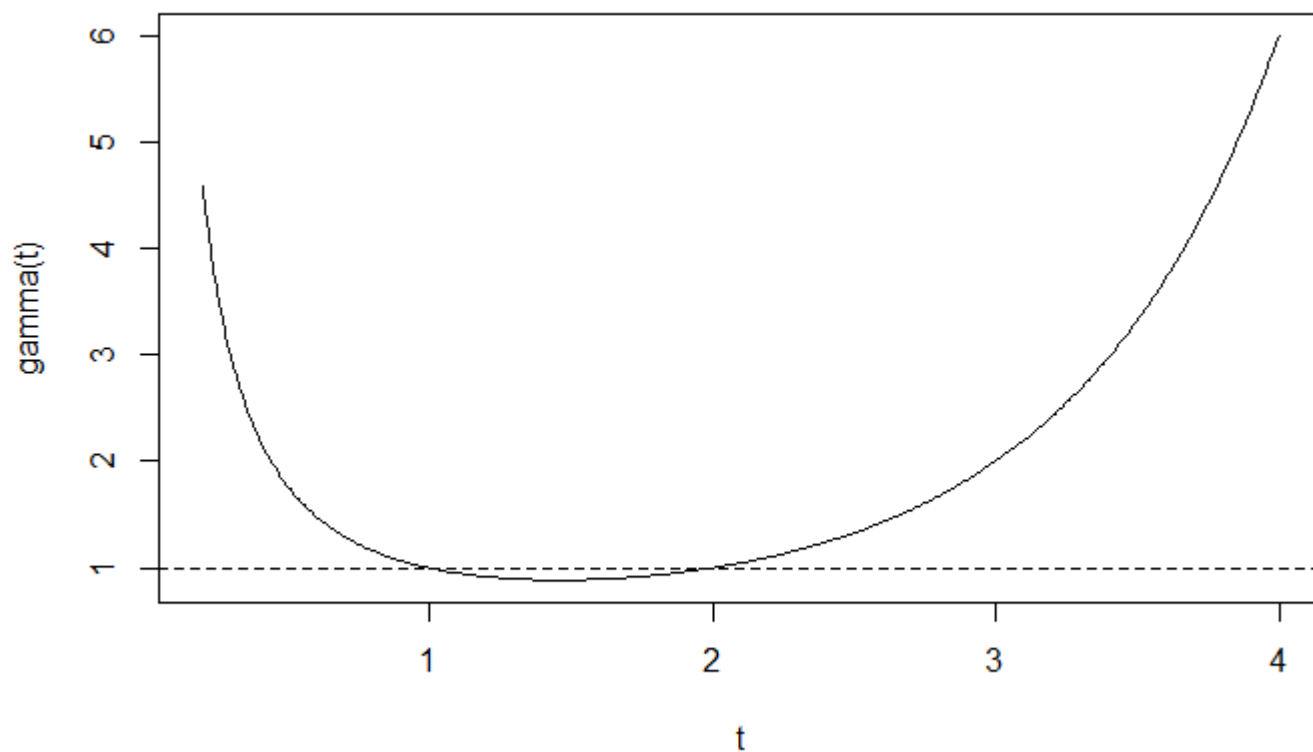
```
plot(x,y3,type="l",ylab="y",main="steep hump")
```



Section 7.1.5

[Hide](#)

```
par(mfrow=c(1,1))  
t <- seq(0.2,4,0.01)  
plot(t,gamma(t),type="l")  
abline(h=1,lty=2)
```



Section 7.1.6

Section 7.1.7

[Hide](#)

```
(1/44.44 - 1/70.59)/(1/0.2 - 1/0.6)
```

```
[1] 0.002500781
```

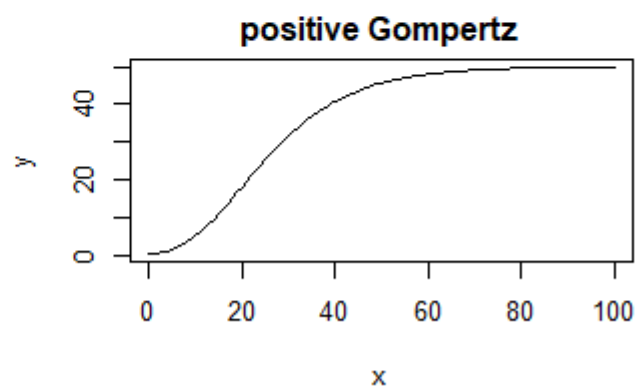
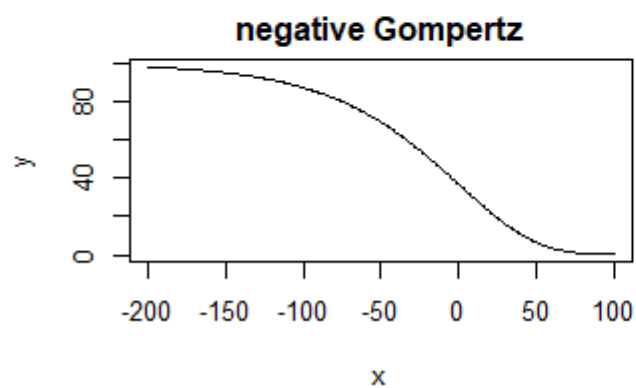
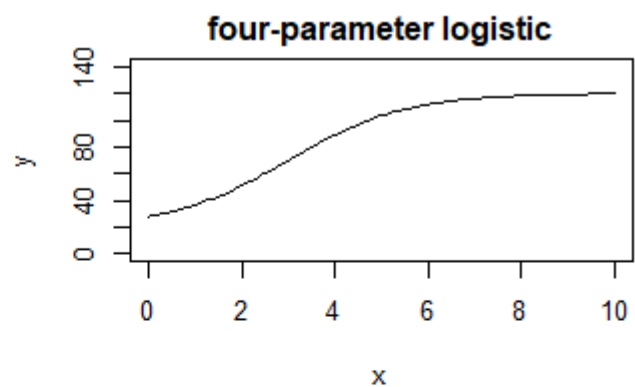
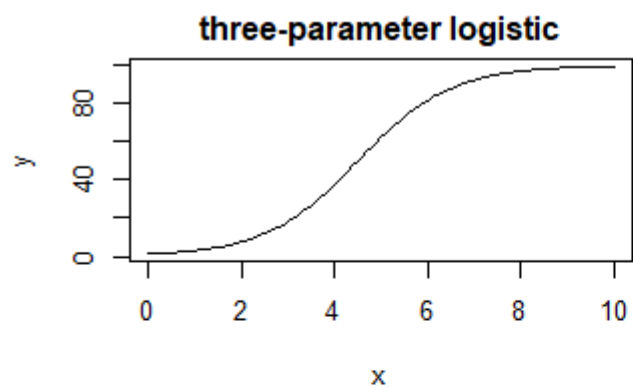
section 7.1.8

[Hide](#)

```
par(mfrow=c(2,2))
x <- seq(0,10,0.1)
y <- 100/(1+90*exp(-1*x))
plot(x,y,type="l",main="three-parameter logistic")
y <- 20+100/(1+exp(0.8*(3-x)))
plot(x,y,ylim=c(0,140),type="l",main="four-parameter logistic")
```

[Hide](#)

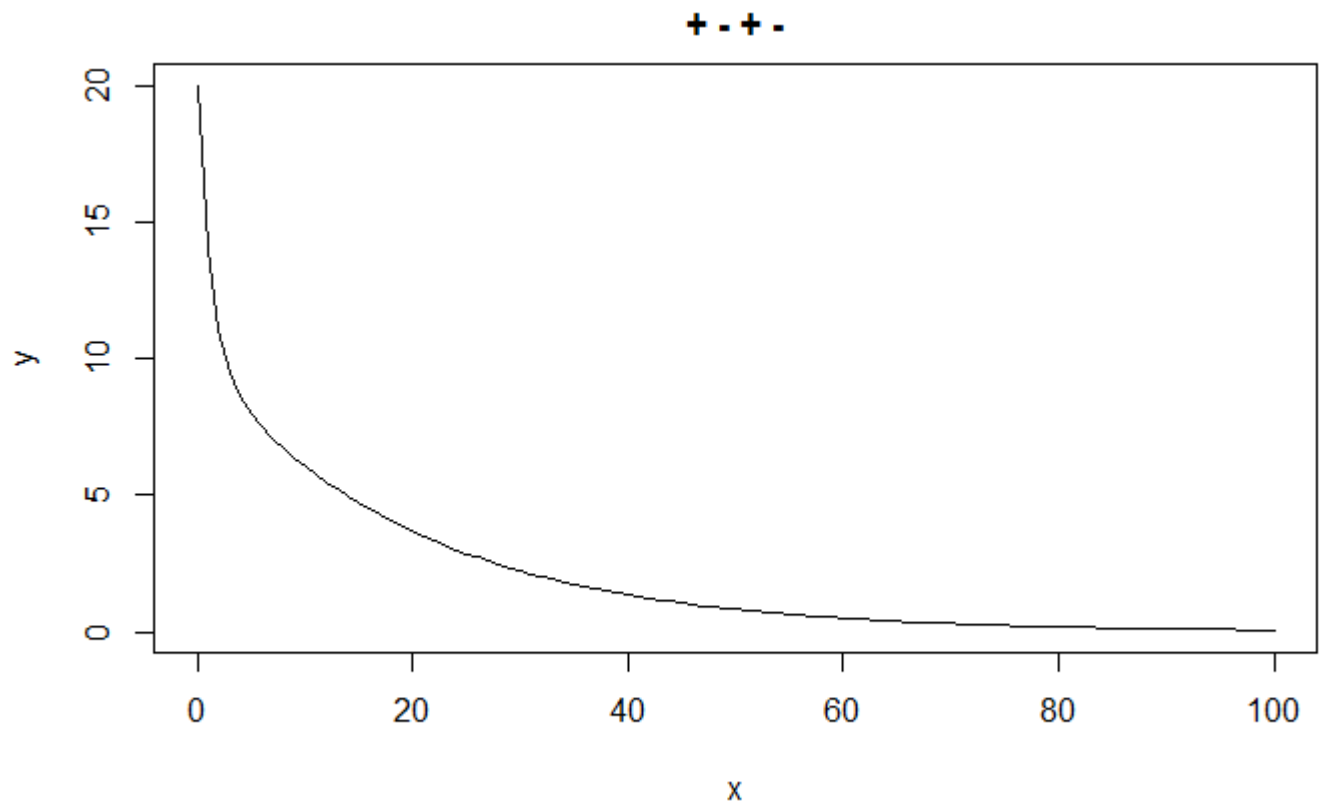

```
x <- -200:100
y <- 100*exp(-exp(0.02*x))
plot(x,y,type="l",main="negative Gompertz")
x <- 0:100
y <- 50*exp(-5*exp(-0.08*x))
plot(x,y,type="l",main="positive Gompertz")
```



Section 7.1.9

[Hide](#)

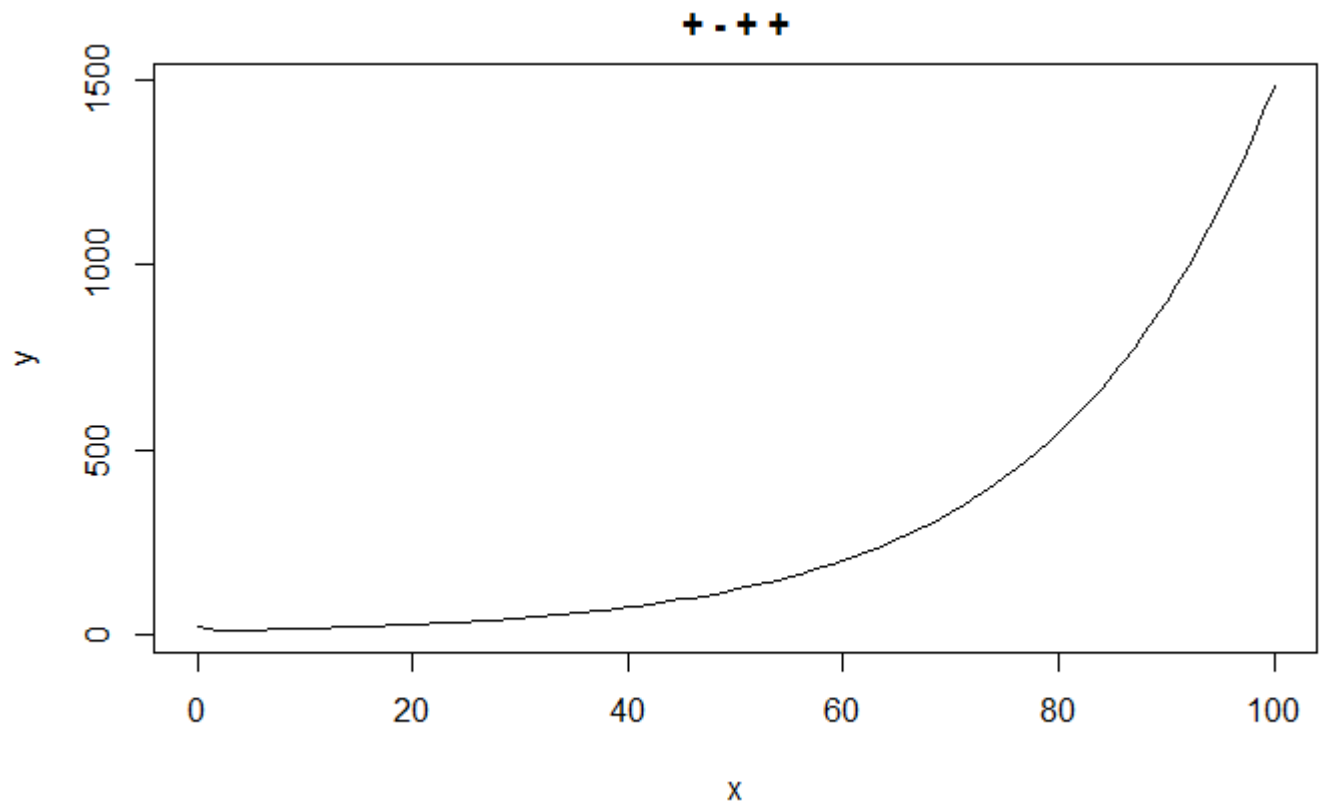
```
#1
a <- 10
b <- -0.8
c <- 10
d <- -0.05
y <- a*exp(b*x)+c*exp(d*x)
plot(x,y,main="+ - + -",type="l")
```



2

[Hide](#)

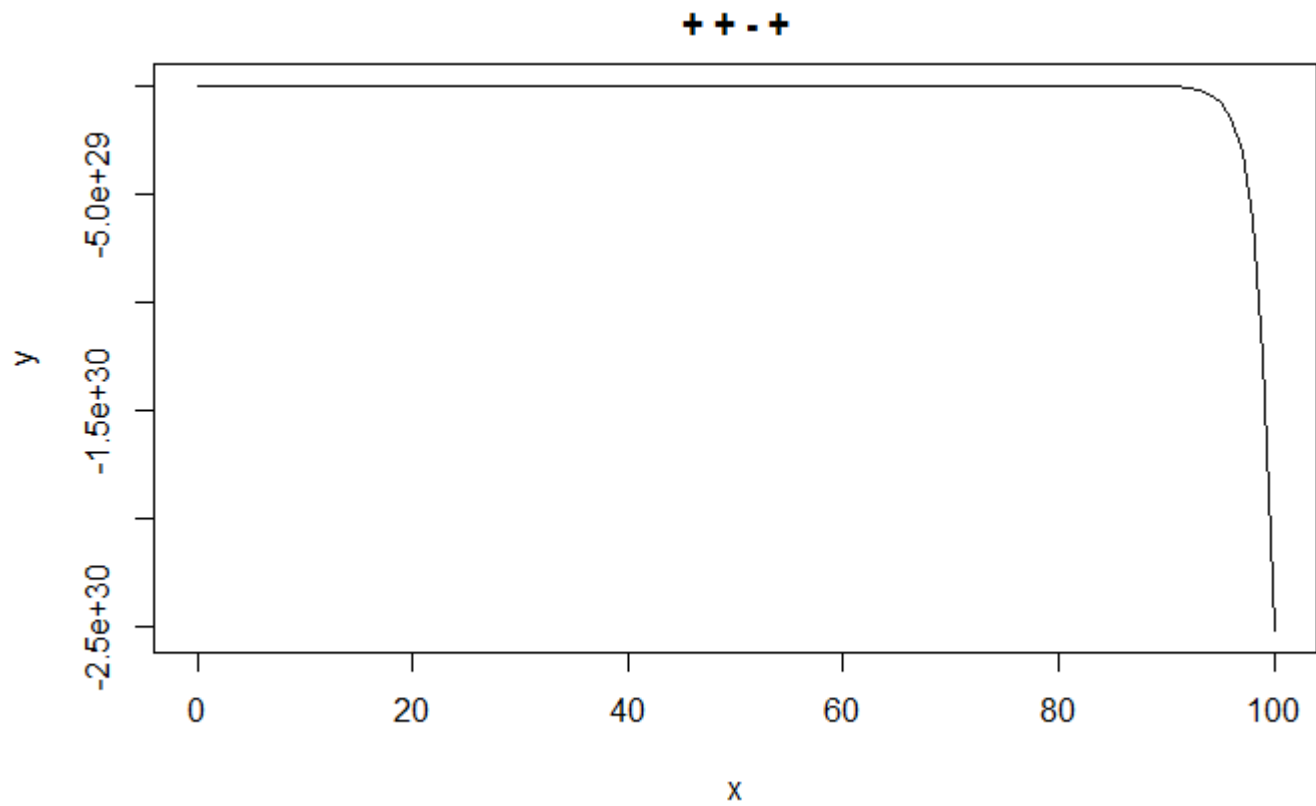
```
a <- 10
b <- -0.8
c <- 10
d <- 0.05
y <- a*exp(b*x)+c*exp(d*x)
plot(x,y,main="+ - + +",type="l")
```



3

[Hide](#)

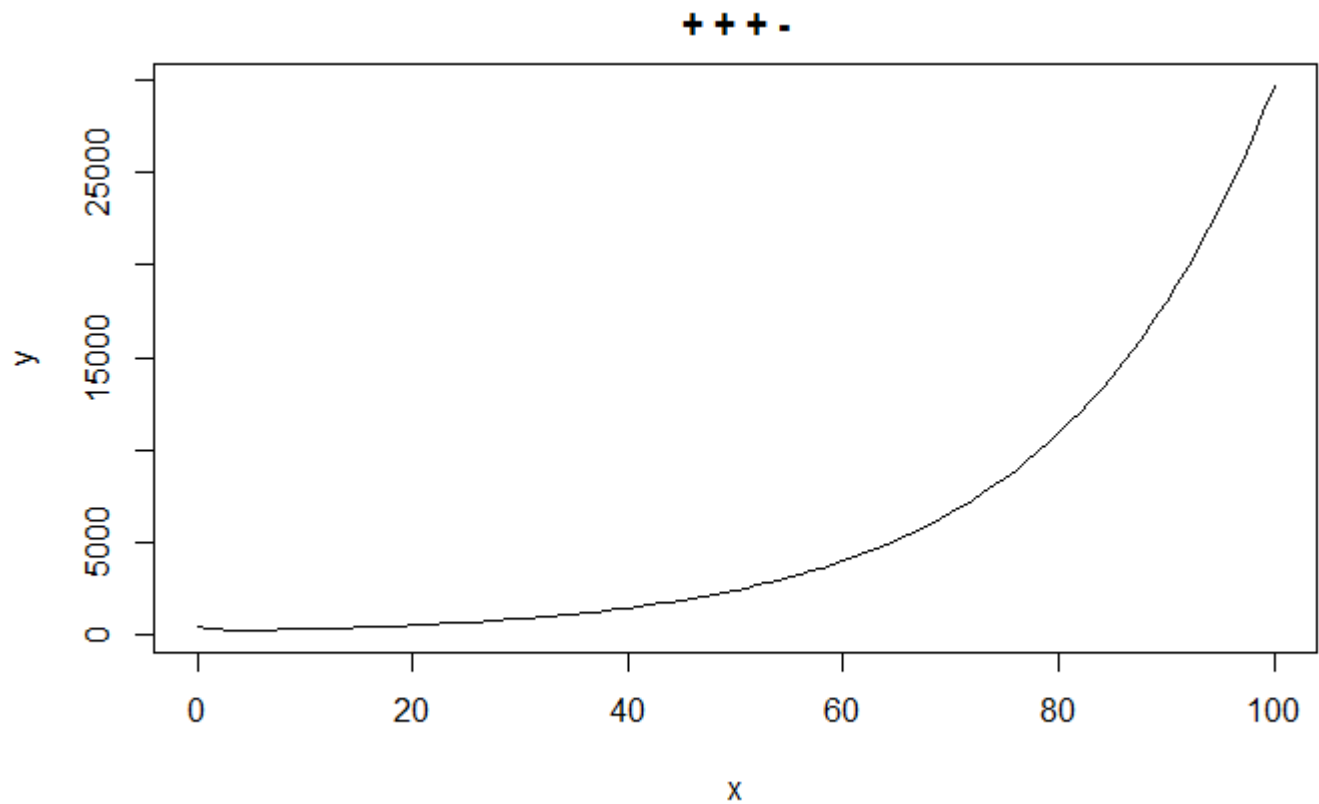
```
a <- 200
b <- 0.2
c <- -1
d <- 0.7
y <- a*exp(b*x)+c*exp(d*x)
plot(x,y,main="+ + - +",type="l")
```



#4

[Hide](#)

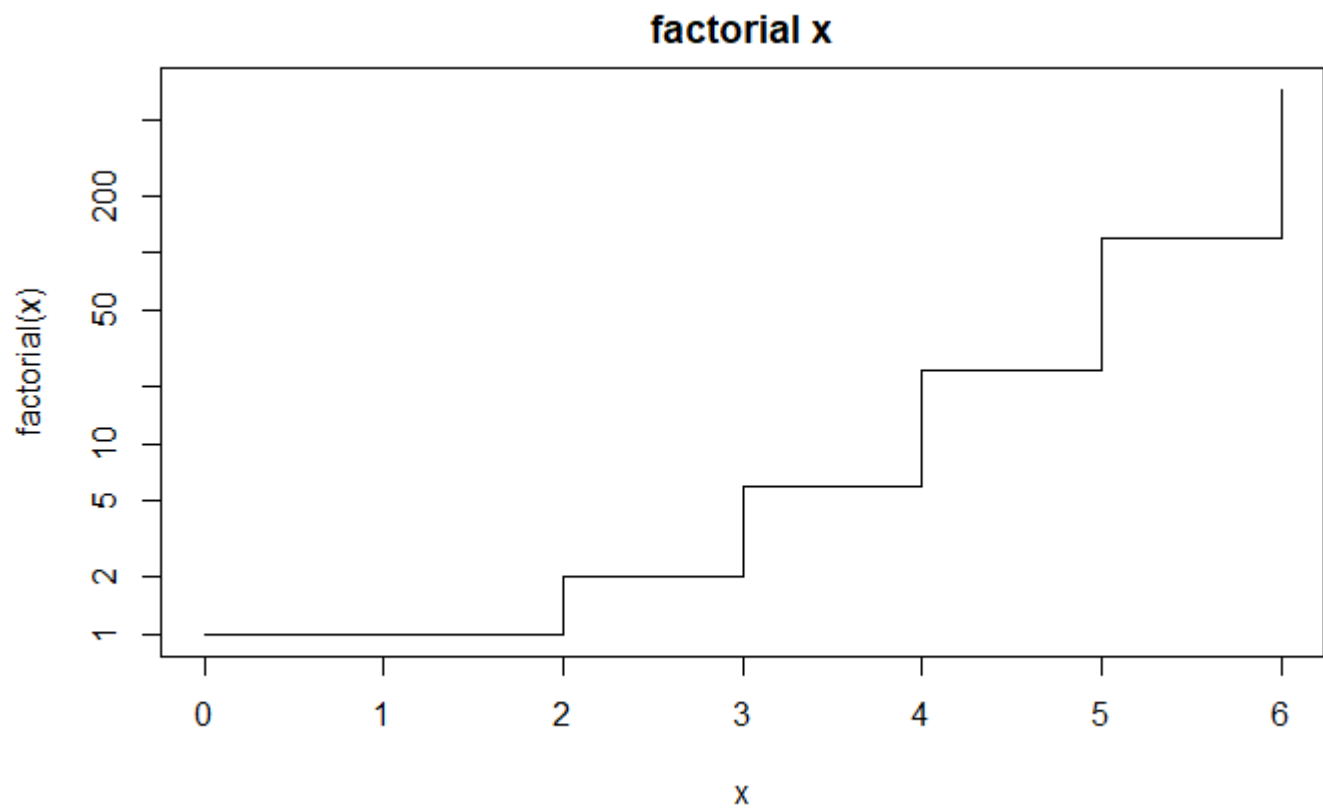
```
a <- 200
b <- 0.05
c <- 300
d <- -0.5
y <- a*exp(b*x)+c*exp(d*x)
plot(x,y,main="+ + - +",type="l")
```



Section 7.2

Hide

```
par(mfrow=c(1,1))
x <- 0:6
plot(x,factorial(x),type="s",main="factorial x",log="y")
```

[Hide](#)

```
choose(8,3)
```

```
[1] 56
```

[Hide](#)

```
plot(0:8,choose(8,0:8),type="s",main="binomial coefficients")
```

